

# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

**FAKULTA ELEKTROTECHNIKY**

**A KOMUNIKAČNÍCH TECHNOLOGIÍ**

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

**ÚSTAV TELEKOMUNIKACÍ**

DEPARTMENT OF TELECOMMUNICATIONS

## ANALÝZA KONVERGOVANÝCH SÍTÍ POMOCÍ SIMULACÍ

ANALYSIS OF CONVERGED NETWORKS BY SIMULATIONS

**BAKALÁŘSKÁ PRÁCE**

BACHELOR'S THESIS

**AUTOR PRÁCE**

AUTHOR

Vít Gardáš

**VEDOUCÍ PRÁCE**

SUPERVISOR

doc. Ing. Vladislav Škorpil, CSc.

**BRNO 2016**



# Bakalářská práce

bakalářský studijní obor **Teleinformatika**

Ústav telekomunikací

**Student:** Vít Gardáš

**ID:** 164269

**Ročník:** 3

**Akademický rok:** 2015/16

**NÁZEV TÉMATU:**

## **Analýza konvergovaných sítí pomocí simulací**

### **POKYNY PRO VYPRACOVÁNÍ:**

Analyzujte integrované konvergované sítě s ohledem na simulaci libovolných problematik v prostředí MATLAB a Simulink. Určete, pro které partie předmětu Vysokorychlostní komunikační systémy je možné s ohledem na zefektivnění výuky použít vhodné modely a simulace. Vybrané simulace realizujte a připravte pro použití ve výuce (minimálně dvě nové laboratorní úlohy).

### **DOPORUČENÁ LITERATURA:**

[1] PUŽMANOVÁ, R. Moderní komunikační sítě A-Z. Computer Press, Brno 2007.

[2] MATLAB, firemní dokumentace. 2004 - 2015

[3] ŠKORPIL, V. Vysokorychlostní komunikační systémy. FEKT, Brno 2014.

**Termín zadání:** 1.2.2016

**Termín odevzdání:** 1.6.2016

**Vedoucí práce:** doc. Ing. Vladislav Škorpil, CSc.

**Konzultant bakalářské práce:**

**doc. Ing. Jiří Mišurec, CSc., předseda oborové rady**

### **UPOZORNĚNÍ:**

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

## **Abstrakt**

Cílem této práce je analyzovat konvergované vysokorychlostní sítě pomocí programu Matlab a Simulink. Určit problematiku, z předmětu Vysokorychlostní komunikační systémy a řešit ji pomocí vytvoření modelu a simulací. Poté vytvořit laboratorní úlohu na toto téma.

## **Klíčová slova**

Matlab, Simulink, Konvergence, síť, QAM, QPSK, PCM

## **Abstract**

The primary purpose of this study is to analyze converged high-speed networks with the usage of Matlab and Simulink. To set problematic area in the field of high-speed networks and solve this problem by creating a model of measurement and simulation of this model. A laboratory exercise is afterwards created on the specific problem.

## **Key words**

Matlab, Simulink, Convergence, network, QAM, QPSK, PCM

GARDÁŠ, V. Analýza konvergovaných sítí pomocí simulací. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2016. XY s. Vedoucí bakalářské práce doc. Ing. Vladislav Škorpil, CSc..

## PROHLÁŠENÍ

Prohlašuji, že svou bakalářskou práci na téma „Analýza konvergovaných sítí pomocí simulací“ jsem vypracoval samostatně pod vedením vedoucího semestrální práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny uvedeny v seznamu literatury na konci práce.

Jako autor uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této semestrální práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

V Brně dne.....

.....

podpis autora

## **PODĚKOVÁNÍ**

Rád bych poděkoval vedoucímu práce panu doc. Ing. Vladislavovi Škorpilovi, CSc. za vedení a cenné rady při zpracování práce.

V Brně dne.....

.....

podpis autora

Výzkum popsáný v této bakalářské práci byl realizován v laboratořích podpořených projektem Centrum senzorických, informačních a komunikačních systémů (SIX); registrační číslo CZ.1.05/2.1.00/03.0072, operačního programu Výzkum a vývoj pro inovace.

# Obsah

Úvod.....	8
1 Matlab a Simulink .....	9
1.1 Matlab.....	9
1.2 Simulink.....	11
2 Historie sítí .....	13
3 Datové sítě.....	14
3.1 Konvergence.....	14
3.2 Paketové sítě, Frame Relay, ATM.....	15
3.2.1 Paketové sítě .....	15
3.2.2 Frame Relay.....	15
3.2.3 ATM .....	16
4 QAM A QPSK .....	18
5 PCM .....	19
6 Laboratorní úloha č.1 .....	22
7 Laboratorní úloha č.2 .....	34
8 Informace pro vyučující k úlohám .....	40
Úloha č.1.....	40
8.1 Odpovědi na doplňující otázky laboratorní úlohy č.1 .....	40
Úloha č.2.....	41
8.1 Odpovědi na doplňující otázky laboratorní úlohy č.2 .....	43
Závěr.....	44
Literatura .....	45
Seznam obrázků .....	47
Seznam tabulek .....	48

# Úvod

Původně byly jednotlivé služby poskytovány odděleně na svých sítích. Telefonie byly provozovány na analogových sítích s přepojováním fyzických okruhů. Datová komunikace probíhala na digitálních sítích s přepojováním paketů, tyto sítě původně sloužily velkým společnostem a výzkumným ústavům ke komunikaci. Sítě byly pomalé a přenášely malé soubory, zprávy atd. Zábavu obstarávalo analogové televizní vysílání. Vývoj technologií a náročnost uživatelů změnili celkovou strukturu těchto služeb. Došlo k integraci všech zmiňovaných služeb na datovou síť, to znamená, že se dostaly do podoby, jak je známe dnes. Sítě jsou velmi výkonné a přenášejí obrovské soubory, videokonference nebo tok dat v podobě videa (video streaming). Mezi „nepřátele“ poskytování těchto služeb patří samotný přenos dat. K tomu, abychom mohli tyto služby využívat je zapotřebí vybudování dostatečně kvalitních infrastruktur, které se nachází ve velkých městech, ovšem na venkově se sítě nenachází na takové úrovni a například při sledování vysílaného streamovaného videa dochází k velkým ztrátám snímků.

Cílem této práce je zvolit problematiku týkající se předmětu Vysokorychlostní komunikační systémy, vytvořit model a simulovat jej v programu Matlab a Simulink, který dále bude použit ve výuce a realizaci laboratorní úlohy. Tyto modely by měly studentům pomoci ověřit si své teoretické znalosti a pochopit nesrovnalosti v dané oblasti.

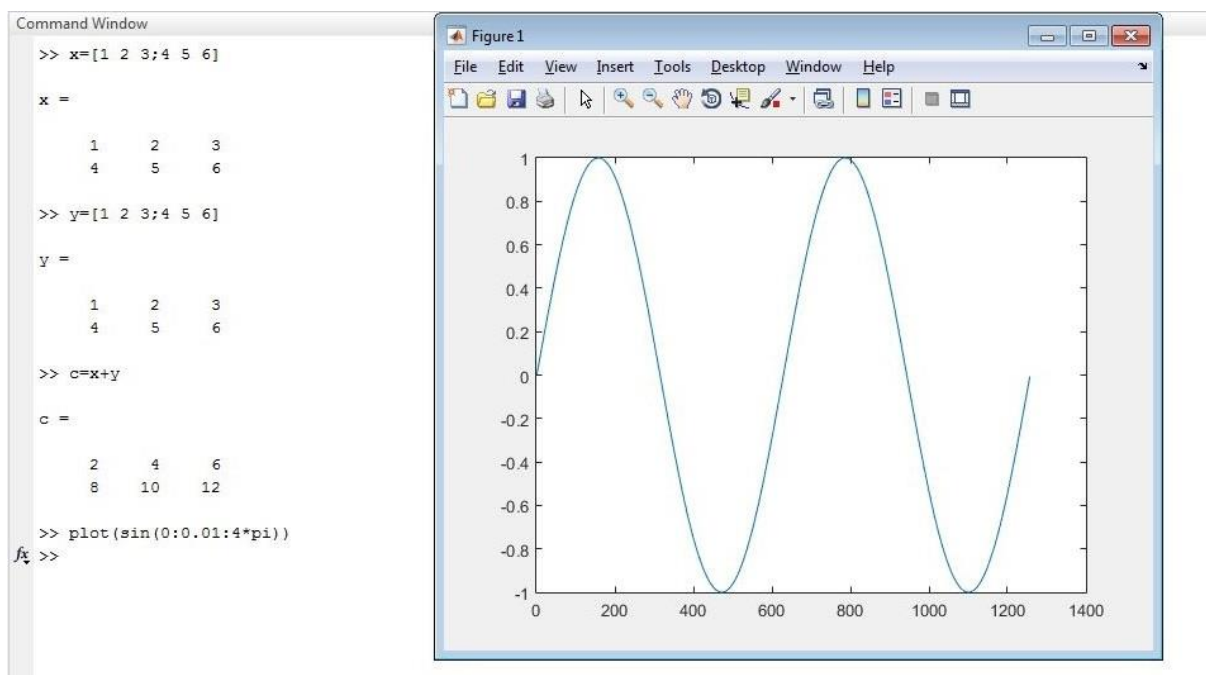


# 1 Matlab a Simulink

## 1.1 Matlab

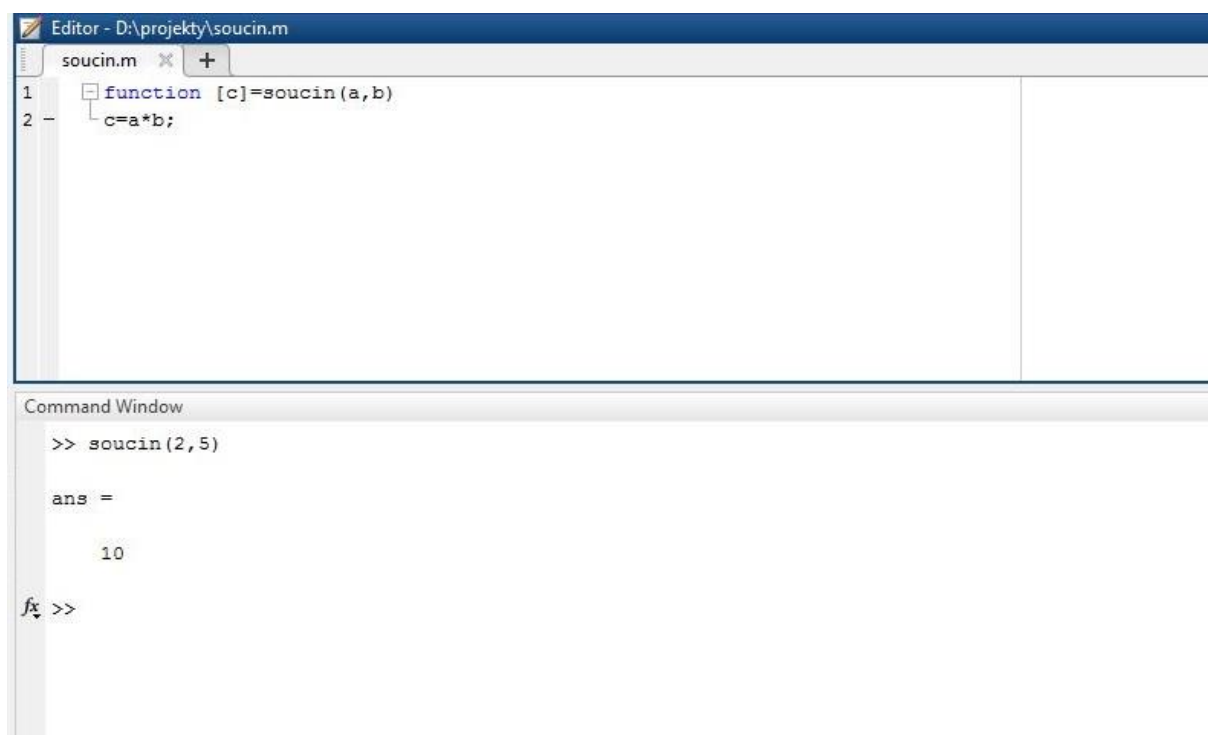
Matlab je programovací prostředí a programovací jazyk od společnosti MathWorks. Program je kompatibilní s operačními systémy Microsoft Windows, Linux a Mac OS. Matlab se využívá pro technické a vědecké výpočty, využívají jej zejména studenti s technickým a ekonomickým zaměřením při výpočtech a simulacích, taktéž jej využívají v praxi různé společnosti, např. HUMUSOFT. Matlab obsahuje ve svých knihovnách funkce, které ve svých výpočtech využíváme, ale taktéž lze do tohoto systému některé specifické tzv. toolboxy (nástroje) do knihovny doinstalovat.

Prostředí programu je velmi intuitivní a přehledné. Po zapnutí se zobrazí hlavní okno Command Window, na levé straně Workspace, který nám usnadňuje práci při tvoření projektů a Current folder, což je zobrazené úložiště, kde můžeme otevírat a spouštět naše uložené projekty. Do Command Window píšeme přímo příkazy, potvrzovány klávesou ENTER, které jsou následně ihned vykonány. Jedná se např. o jednoduchý výpočet matic, grafické zobrazování průběhů, různých funkcí atd., pokud ovšem program vypneme, kód se nám neuloží a nemůžeme s ním pracovat i nadále. Takovéto výpočty a simulace nám slouží zejména k ověření, názorné ukázce a objasnění dané problematiky.



Obr. 1.1: Ukázka jednoduchého výpočtu a zobrazení sinusového průběhu

Pokud ovšem na projektu pracujeme dlouhodobě, budeme se k němu vracet, zdokonalovat ho nebo využívat k opakovaným výpočtům, můžeme jej samozřejmě uložit. K uložení slouží soubory m-file, které obsahují naši funkci. Výsledky výpočtů si zobrazíme v oblasti Command Window pomocí zavolání naší funkce, při zakládání nového skriptu si musíme dát pozor, abychom ho pojmenovali stejně jako naši funkci (Viz. Obr.2), protože pokud bychom ho pojmenovali jinak, funkce by nám nefungovala.



Obr. 1.2: Ukázka práce se souborem m-file

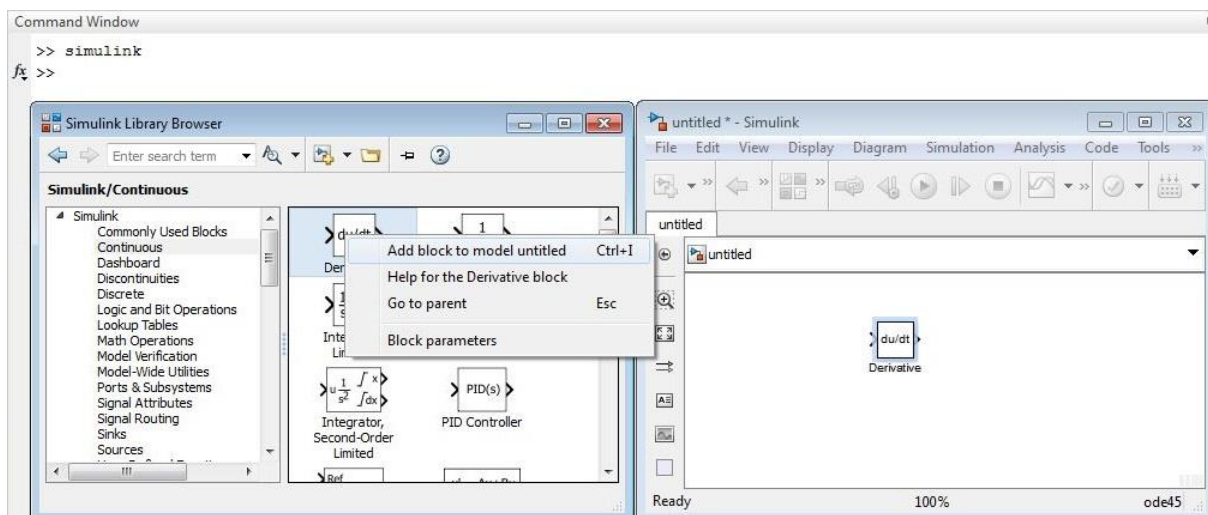
Jestliže pracujeme na obtížnějším úkolu, Jednotlivé skripty s funkcemi se nám mohou prolínat, tudíž v jednom skriptu můžeme volat funkce z jiných skriptů, zobrazovat grafické výsledky atd.

V případě, že bychom zapomněli nebo neznali potřebnou syntaxi, je možnost použít nápovědu, tu najdeme na webových stránkách společnosti MathWorks: <http://www.mathworks.com/help/matlab/>, pomocí této nápovědy bychom měli vyřešit náš problém. Pokud nemáme připojení k internetu, tak můžeme použít také offline nápovědu, kterou najdeme přímo v programu. Tuto offline nápovědu zobrazíme vložením příkazu *help* do Command Window.

## 1.2 Simulink

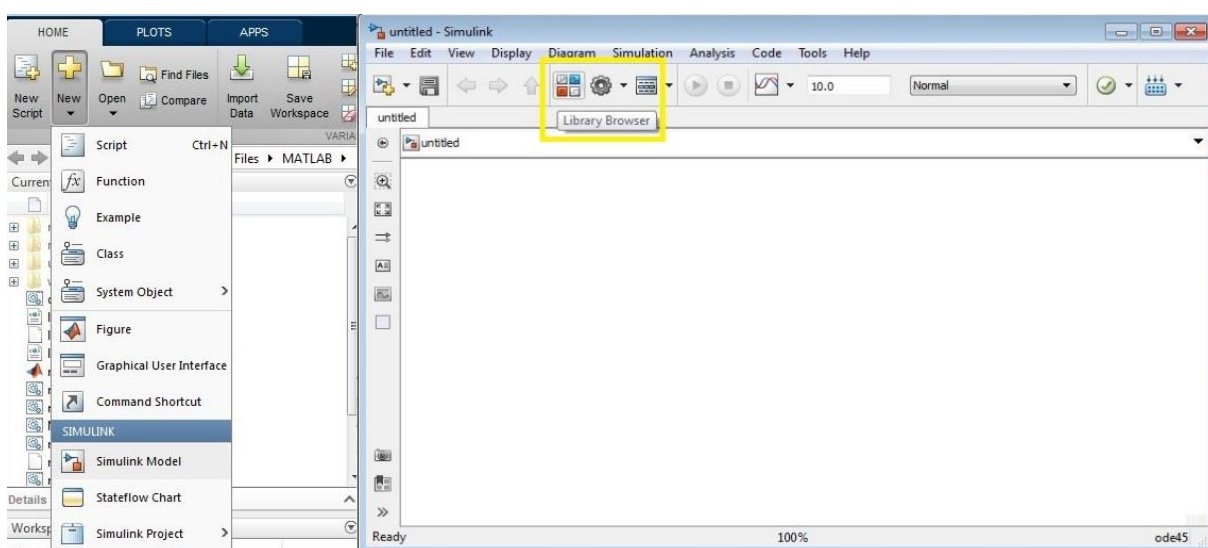
Simulink je softwarový balíček pro modelování, simulaci a analýzu dynamických systémů. Je to nástavba programu Matlab. Původně to byla jen jednoduchá funkce v knihovně, která sloužila k simulaci jednoduchých systémů. Nyní tento nástroj obsahuje bloky, které představují dynamické systémy. Měnit nastavení či hodnoty různých parametrů lze velmi snadno a tyto změny se nám okamžitě projeví ve výsledcích simulace. Složitější systémy vytvoříme spojením těchto předpřipravených bloků z knihovny simulinku. Pokud v knihovně nenalezneme blok, který realizuje námi potřebný systém, lze stejně jako u matlabu doinstalovat některé specifické toolboxy. Musíme si ovšem dát pozor, aby byly toolboxy kompatibilní s aktuální verzí našeho programu. Pokud potřebný systém nenajdeme ani v těchto doinstalovaných knihovnách, můžeme si jej vytvořit tak, aby nám vyhovoval a plnil vyžadované funkce.

Do prostředí programu Simulink se dostaneme tak, že do Command Window v Matlabu napíšeme *simulink* a potvrdíme enterem, tak jako jsme zvyklí z používání Matlabu. Po chvilce se zobrazí prohlížeč knihoven Simulinku, zde jsou bloky rozděleny podle funkcí do jednotlivých skupin. Potřebný blok vyhledáváme pomocí tohoto rozdělení a nebo pomocí vyhledávače přímo v prohlížeči v levém horním rohu. Na blok, se kterým chceme pracovat, klikneme pravým tlačítkem myši a zvolíme *Add block to model untitled* (Viz. Obr. 1.3), zobrazí se nám už konkrétní prostředí Simulinku a na pracovní plochu se nám přidal zvolený blok, tímto způsobem přidáme další prvky potřebné pro náš systém. Tyto bloky sestavíme a spojíme na již zmiňované pracovní ploše a můžeme přejít k nastavení a úpravám systému.



Obr. 1.3: Ukázka prostředí a knihovny Simulink

Druhou možností zobrazení prostředí Simulinku je v Matlabu kliknutím na ikonu *New* v nabídce *HOME* vlevo nahoře. Takže klikneme na tlačítko *New* a z nabídky, která se zobrazila, zvolíme *Simulink Model*, tím jsme spustili Simulink a můžeme začít pracovat. Jediný rozdíl mezi předešlou variantou je zobrazení přímo prostředí Simulink, tudíž pracovní plochy, kam vkládáme naše bloky systému. Zde prohlížeč knihoven zobrazíme kliknutím na ikonu *Library Browser*, kterou najdeme v hlavní nabídce programu. Následně vybíráme a vkládáme bloky stejně jako v prvním případě.



Obr. 1.4: Druhá možnost zobrazení prostředí Simulink

## 2 Historie sítí

V minulosti v 70. letech se sítě rozdělovaly zejména z hlediska technologií, zpracování a jejich užitečnosti. Jednalo se tedy o sítě pro přenos hlasové komunikace, datové komunikace a zábavu. K hlasové komunikaci se využívaly analogové sítě s komutací okruhů nebo radiové sítě. Zábavu zajišťovaly zejména rozhlasové a televizní sítě. Datová komunikace v této době byla velmi mladá a dostala se do tohoto rozdělení jako poslední. Sloužila zejména k propojení terminálů k nadřazenému počítači, buďto přes sériové spoje nebo pomocí modemu, který se používal na větší vzdálenosti.

V 80. letech toto rozdělení do tří táborů zůstalo stejné, ovšem docházelo k jednotlivým inovacím systémů. U hlasové komunikace se začínají digitalizovat ústředny (střed sítě), ale okraje sítě neboli uživatelé stále využívají sítě analogové. Taktéž se začínají vyvíjet mobilní komunikace. Z hlediska zábavy se začínají využívat kabelové sítě a například Bulletin board systém (BBS), což je systém elektronických nástěnek. K připojení k tomuto systému uživateli stačil počítač s vhodným modemem a telefonní číslo, aby se mohl na stanici BBS dovolat. Pro představu tento systém fungoval jako dnešní diskusní fóra, lidé zde získávali informace a také je zde zanechávali. Největší rozvoj zažily datové komunikace. Ve velké míře se rozšiřuje přenos souborů, elektronické pošty a přihlašování na dálku. Začínají se využívat lokální sítě ke spojení osobních počítačů se servery. Ke spojení se využívá pevné metalické médium a nejčastější je topologie sběrnicová a kruh neboli Token Ring. U rozlehlých sítí se objevuje systém přepínání paketů, např. u sítě Arpanet, která je páteří internetu.

V 90. letech dochází rovněž k sestavování a zdokonalování pravidel komunikace. Tento souhrn norem je velmi důležitý, abychom mohli po síti komunikovat i s různými zařízeními a naše spojení proběhlo v pořádku. Tento celý problém byl rozdělen na dílčí problémy a tyto dílčí problémy řeší jednotlivé vrstvy. Toto všechno řeší ISO/OSI model (nyní se využívá model TCP/IP), který zahrnuje záležitosti přenosových medií, modulací až po konkrétní aplikace v našem operačním systému. Tento normalizační model byl přijat v roce 1984.

Cílem bylo sloučit tyto jednotlivé možnosti a funkce, tak aby vznikl nějaký integrovaný systém, který by to umožnil. Což už se v dnešní době stává realitou, kdy můžeme pomocí datové komunikace nejenom odesílat emaily a malé soubory, ale také provádět telekonference a videokonference za pomoci různých programů. Dále je také možnost sledování televizního vysílání ve formě streamovaného videa na internetu. K tomuto všemu,

ale potřebujeme vysokorychlostní síť s obrovskou propustností dat, abychom takový přenos mohli uskutečnit. [15]

## **3 Datové síť**

### **3.1 Konvergence**

Konvergence je chápána jako splynutí neboli integrování oddělených přenosových infrastruktur. Jedná se zejména o datové a telekomunikační síť. Jak jsem se již zmiňoval, telekomunikační síť byly analogové a byly založeny na přepojování okruhů. Byly spolehlivé a dle mého názoru ne tak pokrokové a ambiciózní jako datové síť. Takže vedle hlasové komunikace se zde vyvíjela i komunikace datová, která se zdokonalovala poměrně rychle. Datové síť měly původně sloužit k jiným účelům a poskytovat jiné služby. O tyto systémy se zajímaly zejména velké společnosti, které tyto funkce potřebovali a využívali je. Široká veřejnost se do styku s těmito technologiemi vůbec nedostala. A to je zřejmě hlavní důvod toho, proč docházelo k takovému rozvoji těchto systémů. Mezi poskytovateli těchto služeb byla velká rivalita a snaha nabídnout to nejlepší, což byl jeden z motorů vývoje.

Telefonie se tedy dostala i do zájmu datových sítí, což bylo pro uživatele velmi přínosné, zejména z důvodu finančního. Tato záležitost se dostala tak daleko, že společnosti nyní využívají k pracovním telekonferencím nebo videokonferencím jen datové síť. Výhod to má spousta, firmy nemusí spravovat více infrastruktur, ale stačí jim jedna komplexní infrastruktura, která dostatečně poslouží jejím požadavkům. Což má za následek menší finanční rozpočty.

## **3.2 Paketové sítě, Frame Relay, ATM**

### **3.2.1 Paketové sítě**

Paketová síť fungovala na odesílání paketů o různé velikosti. Tento přenos využívala síť X.25, která byla definována a používána ještě před vznikem normalizačního modelu ISO/OSI. Po vytvoření modelu byly parametry a normy upraveny tak, aby její síť splňovala. Ke komunikaci je využíváno PVC i SVC. PVC značí Permanent Virtual Circuit, což je vytvoření pevného virtuálního okruhu, který je poskytován na delší dobu a neruší se okamžitě po dokončení komunikace. SVC je přepínaný virtuální okruh, u kterého dojde k sestavení komunikace, průběhu komunikace a po ukončení spojení se okruh zruší a při novém navázání vytvoří nový. Výhodou tohoto systému je vcelku solidní spolehlivost a cenová dostupnost. Ovšem nevýhodou je maximální rychlost přenosu 64 kbit/s, což je v dnešní době opravdu málo a o nějakém multimediálním přenosu, proudového vysílání zde nemůže být ani řeč. Také docházelo k zahlcení a zpomalení sítě po připojení více uživatelů.

### **3.2.2 Frame Relay**

Snaha zrychlit přenos přinesla přepojování rámců. Tento způsob funguje na přenosu dat pomocí rámců o proměnné délce. Rychlost přenosu se zvýšila tím, že se kontrolou chybovosti nezabývaly prvky sítě v průběhu komunikace. Tuto funkci zastávala až koncová zařízení po přijetí dat. Oproti síti s přepojováním paketů se odezva sítě snížila z 200ms až na 20ms, což byl obrovský skok. Přenosová rychlost se pohybuje od 64 kbit/s až do 2048 Mbit/s. Tento komunikační systém je díky svým vlastnostem vhodný například pro propojování lokálních sítí. Pro spojení používá stejně jako síť X.25 pevné a přepojované virtuální okruhy (PVC, SVC).

Frame Relay má poměrně dobrou výkonnost a možnost upravovat šířku pásma. Jeho nevýhodou je ovšem proměnná délka rámců, což není vhodné pro přenos videa a celkově pro proudový přenos dat v reálném čase.

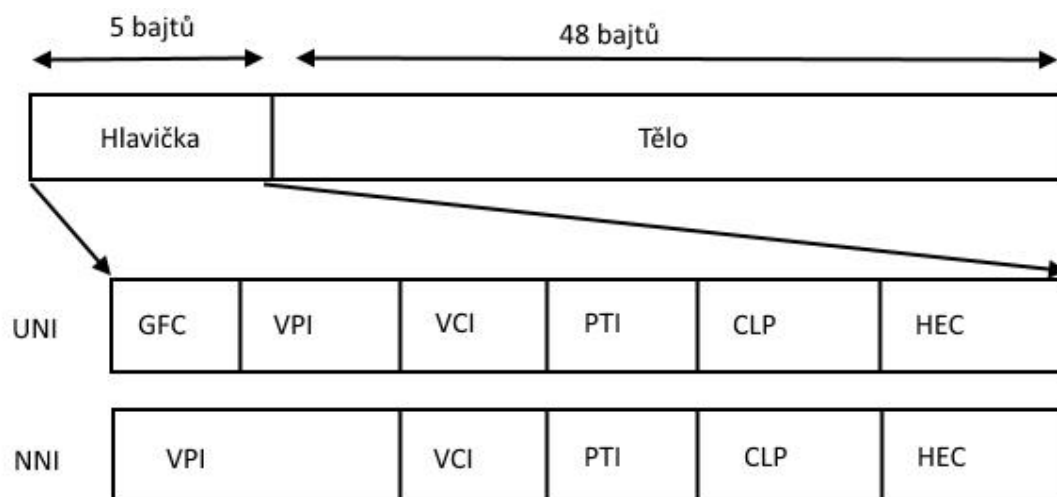
### 3.2.3 ATM

Později přišla nová technologie přepojování buněk, protože se zvyšovaly nároky na šířku pásma a kvalitu služeb. Technologii přepojování buněk využívá asynchronní přenosový režim neboli ATM (Asynchronous Transfer Mode). ATM síť je složena z rychlých prepínačů (switchů), přenosových médií, které přenáší data mezi jednotlivými prepínači. ATM se zabývá normalizací rozhraní mezi uživatelem a sítí a také postavení jednotlivých prepínačů mezi sebou.

Oproti paketům a rámcům mají ATM buňky pevně danou velikost a jsou vhodnější pro přenos videa v reálném čase. Stanovená velikost buňky je 53 bajtů. Těchto 53 bajtů je rozděleno na 5 bajtů pro záhlaví a zbylých 48 bajtů je tělo buňky, které obsahuje data. Při přenosu sítí se na každém prepínači zkontroluje hlavnička, poté upraví a buňka se posune dále v síti. Využívají se dva formáty buněk a to UNI a NNI. Každý z nich se využívá za jiným účelem. UNI se využívá u přístupového protokolu, což znamená spojení sítě s účastníkem. NNI slouží uvnitř sítě v síťovém protokolu, kde dochází ke komunikaci mezi prvky sítě (prepínače), nikoli ke komunikaci s účastníkem. Tyto formáty buněk se také liší ve složení záhlaví, základem u obou formátů buněk je řídicí část, což je PTI A CLP, dále HEC, která obsahuje zabezpečení a liší se v identifikaci, kterou zastupují dvě části VPI a VCI, UNI obsahuje ještě GFC, které slouží k řízení datového toku (viz. Obr.3.1).

Buňky se přenáší v multiplexním toku, což umožňuje jejich náhodný výskyt (buňky se periodicky neopakují) a upřednostňování jednotlivých služeb. Zde dochází k nastavení QoS (Quality of Service), toto nastavení je velmi důležité, protože aplikace, které pracují v reálném čase, jako je například videokonference, požadují rychlý přenos a co nejmenší zpoždění. Takže již zmiňovaná videokonference bude mít vyšší prioritu, než například stahovaný soubor, u kterého jde zejména o to, aby data přišla nepoškozena a všechna. [14]





Obr. 3.1: ATM buňka

Při vytváření spoje mezi účastníky systém sleduje stav sítě, zda jsou požadované přepínače k dispozici nebo zda mají požadovanou kvalitu, např. pro přenos videa. Pokud je síť přetížená, může signalizaci na vytvoření spoje odmítnout.

Celkově jsou ATM sítě v dnešní době velmi využívány, jejich největšími výhodami jsou možnosti nastavení šířky pásma dle potřeby, zasílání dat dle priorit a využití jak ve WAN, tak v LAN sítích. Tato technologie je ovšem složitější, než předešlé varianty a je tudíž ekonomicky náročnější.

## 4 QAM A QPSK

Tyto modulace mají mnohé využití, mezi které patří i digitální kabelové televizní vysílání a satelitní vysílání.

Původně bylo kabelové vysílání pouze analogové a sloužilo jen k přenosu televizního vysílání. Obraz byl přenášen po koaxiálním kabelu. Tyto služby disponovaly širokou škálou televizních stanic, což u běžného pozemního analogového vysílání nebylo běžné, a také dobrou kvalitou, která byla dosažena odolností při přenosu vůči průmyslovému rušení i vůči počasí. Se stálým pokrokem technologií se z analogového vysílání přešlo na vysílání digitální, které přineslo možnosti vysílání více programů a také ve vyšší kvalitě HD (High definition). V jádru sítí neboli páteři koaxiální kabel nahradil optický kabel, který má lepší vlastnosti, než původní koaxiální kabel. Sítě se o hodně zrychlily, začali se využívat k datovým přenosům a zařadily se mezi vysokorychlostní. Nyní velké společnosti nabízejí služby, jak televizního vysílání, tak i poskytnutí připojení k internetu.

QAM i QPSK jsou vícestavové modulace, které se v různých stavech liší ve fázi a amplitudě. U každé varianty těchto modulací se mění počet bitů přenesených na jeden symbol (viz. Tab. 4.1).

Tab. 4.1: Rozdělení modulací

Modulace	Počet bitů na symbol
QPSK	2
16-QAM	4
64-QAM	6

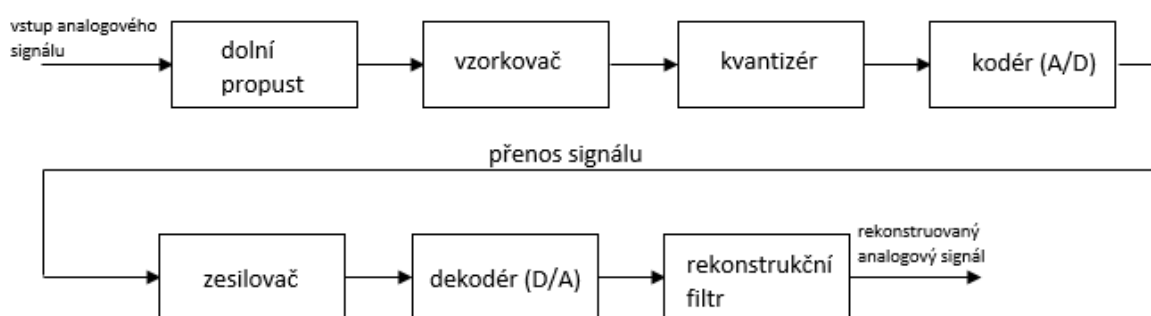
Vícestavové modulace vyšších řádů mají výhodu v rychlosti v přenosu dat po síti, ovšem tato skutečnost je ovlivněna také vyšší chybovostí při přenosu, než u nižších řádů těchto modulací. Je to jednoduše způsobeno tím, že čím více stavů máme, tím je větší pravděpodobnost, že chyba vznikne. Nejrobustnější ze zmiňovaných modulací je QPSK, která má jen 4 stavy. Vyšší řády těchto modulací jsou opatřeny samoopravnými kódy, které představují redundantní bity, kterými se prokládají užitečná data.

Jednotlivé konstelační diagramy a závislost vzniku chyb na šumu si představíme v laboratorní úloze 1.

## 5 PCM

Tato pulzní kódová modulace se řadí mezi digitální modulace v základním pásmu a je hojně využívána při převodu analogového signálu na digitální v mnoha systémech. Zejména je to využití při hovorových přenosech.

Proces modulace se rozděluje do tří základních kroků, jimiž jsou: **Vzorkování**, **Kvantování** a **Kódování**. Princip celé modulace je znázorněn v blokovém schématu (viz. Obr. 5.1).



Obr. 5.1: Blokové schéma PCM

Vstupní analogový signál prochází přes dolní propust, kde dojde k odstranění vyšších frekvencí, a tím se připraví pro proces vzorkování. Dále se ve vzorkovači vytvoří jednotlivé vzorky, které jsou dané zvolenými časovými intervaly, čímž vznikne navzorkovaný PAM signál. Následně proběhne kvantování, ve kterém je průběh rozdělen na určitý počet kvantizačních hladin, těm odpovídá i diskrétní hladiny vzorků. Každou hladinu lze vyjádřit pomocí několika symbolů, které se označují jako kódové slovo, tento cyklus již náleží samotné části kódování. Tyto stavy popisují skupiny bitů, které mohou nabývat hodnot logické 1 nebo logické 0. Již zmiňované kódové slova mohou zastupovat více kvantizačních hladin, počet hladin je určen jednoduchým vztahem, kde je zásadní počet použitých bitů.

$$L = 2^n \quad (5.11)$$

Ve vztahu 5.11 se uvádí počet možných hladin  $L$ , které mohou být zastoupeny kódovým slovem. Mocnina  $n$  určuje počet bitů ve slovu. Ze vztahu lze také určit počet bitů z množství hladin (viz. 5.12).

$$n = \log_2 L \quad (5.12)$$

Používají se různé způsoby kódování, u hovorového přenosu je nejobvyklejší kodek s 256 kvantizačními hladinami. Z předchozích vztahů je zřejmé, že 256 hladinám odpovídá 8 bitů využitých na jednu hladinu.

Po přenosu musí být signál na přijímací straně zesílen v zesilovači. Dále dochází k úpravě signálu pomocí dolní propusti a poté už je možné provést dekódování. Dekódování probíhá takovým způsobem, že se převádí kódová slova, která určují jednotlivé kvantizační hladiny na spojitý signál, následně projde přes rekonstrukční filtry a na výstupu tohoto bloku už je původní rekonstruovaný signál.

V praxi dochází k tomu, že kvantovaný (diskrétní) signál nedokáže totožně sledovat spojitý signál, což způsobuje kvantizační šum. Tento šum ovlivňuje a zkresluje výsledná přijatá data. Aby k tomuto jevu nedocházelo, využívá se robustní kvantování, což zvýší efektivitu modulace a to náhodným kvantizačním krokem. Vzorkovaný signál prochází první dynamickým kompresorem s nelineární charakteristikou a následně lineárním kvantizérem. Koncovým bodem je expander, který svou charakteristikou doplní již kvantovaný signál.

Pulzní kódová modulace je taktéž využívána u modulací signálů na hudební CD. Používají se různé verze těchto modulací s odlišnými přenosovými rychlostmi. Tyto alternativy jsou zaznamenány v Tab. 5.1.[19]

Tab. 5.1 Shrnutí využívaných modulací u hudebních CD

Počet bitů	Frekvence	Přenosová rychlost
16 b	44,1 kHz	88 kB/s
24 b	96 kHz	288 kB/s
24 b	192 kHz	576 kB/s
24 b	384 kHz	1,152 MB/s
32 b	384 kHz	321,5 MB/s

Všechny tyto teoretické znalosti PCM modulací, které jsme zde probrali, si budete moci ověřit simulací v programu Simulink, v laboratorní úloze č.2.

## **6 Laboratorní úloha č.1**

Laboratorní úloha je vypracována a odsimulována v Matlabu verze R2015a, u starších verzí není jistota, že simulace podle návodu bude funkční, či bude pracovat správně.

### **Úloha č. 1 – Simulace modulací 16-QAM, 64-QAM a QPSK**

#### **Zadání a cíl úlohy**

Seznamte se s prostředím Matlab a Simulink. Ověřte své teoretické znalosti v oblasti modulací 16-QAM, 64-QAM a QPSK pomocí simulace. Simulujte modulaci, přenos a demodulaci signálu dle návodu.

1. Sestavte schéma zapojení modulace 16-QAM
2. Sestavte schéma zapojení modulace 64-QAM
3. Sestavte schéma zapojení modulace QPSK
4. U všech zapojení si zaznamenávejte naměřené hodnoty do tabulky v programu Microsoft Excel a následně jednotlivé metody porovnejte a výsledky ukažte vyučujícímu.

#### **Vybavení pracoviště**

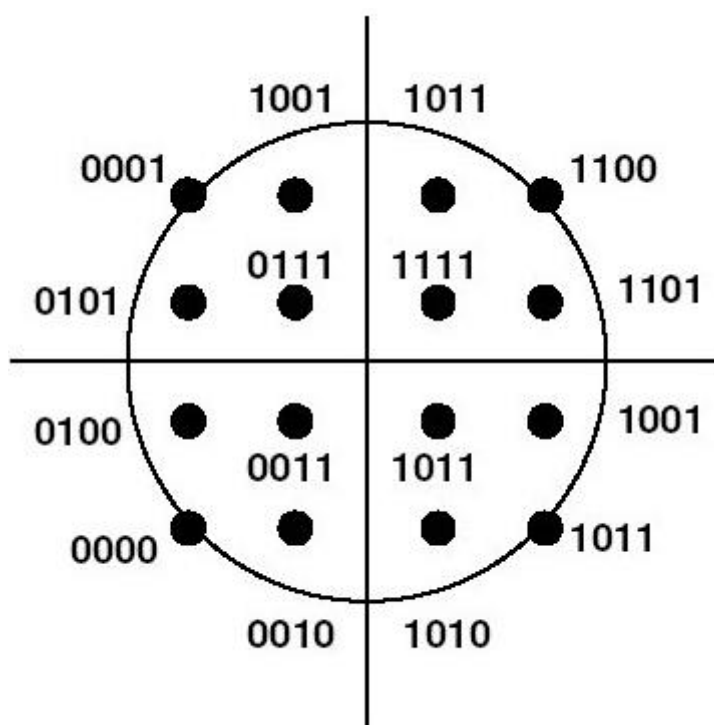
- PC
- program MATLAB
- program Microsoft Excel

# Teoretický úvod

## Modulace QAM a QPSK

Kvadrurní amplitudová modulace (QAM, Quadrature Amplitude Modulation) je digitální a vícecestavová. Tyto modulace jsou hojně využívány u kabelového digitálního vysílání. Nejvíce používané jsou 16-QAM, 64-QAM a 256-QAM, v úloze se budete zabývat šestnácti stavovou a šedesátičtyř stavovou modulací.

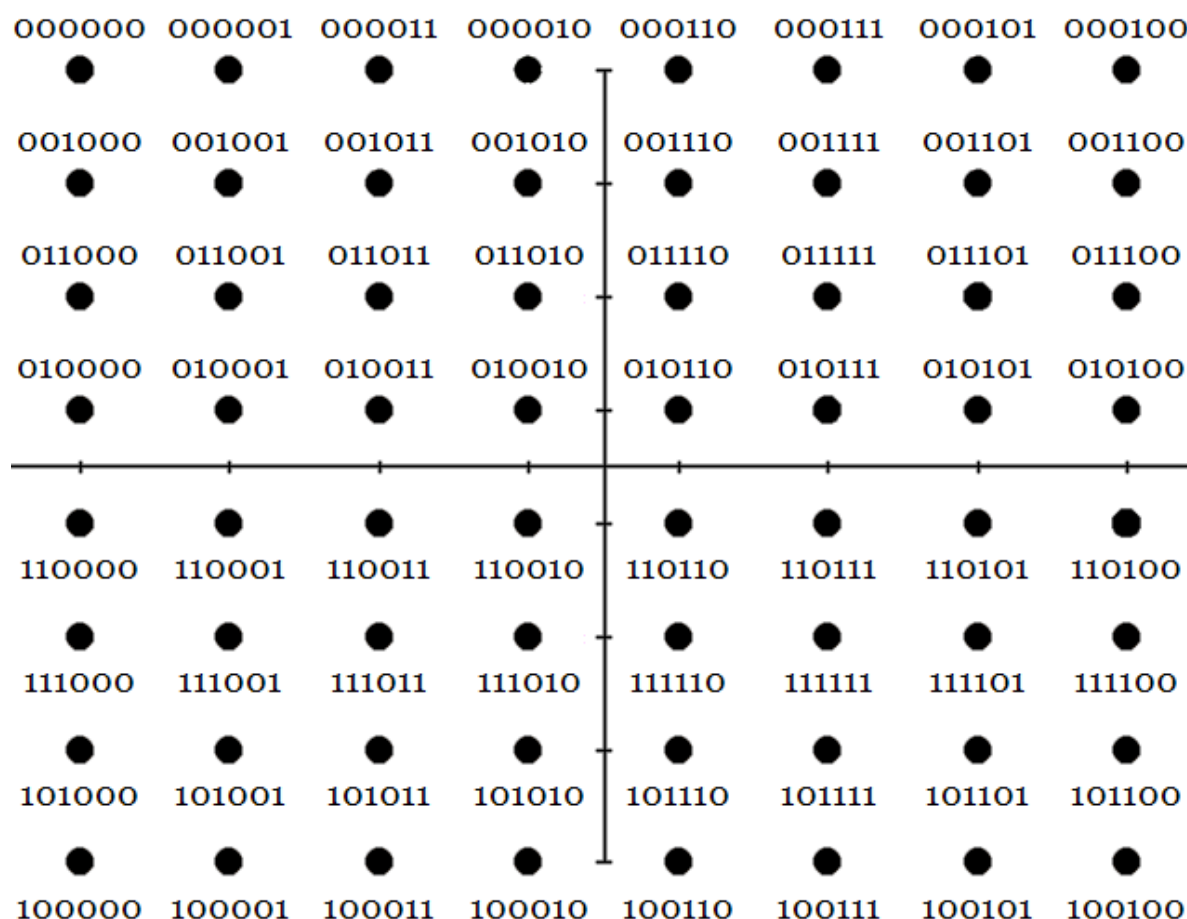
Těchto 16 stavů se odlišuje ve fázi i v amplitudě. Při přenosu modulované subnosné se přenáší 4 bity, což vyplývá z názvu ( $2^4 = 16$  stavů). Tato metoda je také méně náchylná na vznik chyb při přenosu, než 64-stavová modulace. Ovšem čím více stavů při modulaci je použito, přenášené informace mohou dosáhnout vyšší přenosové rychlosti.



Obr. 6.1: Konstelační diagram 16QAM

Modulace 64-QAM má 64 stavů, které se taktéž jako u předešlé metody liší ve fázi i amplitudě. Při přenosu modulované subnosné dochází k přenosu 6 bitů ( $2^6 = 64$  stavů).

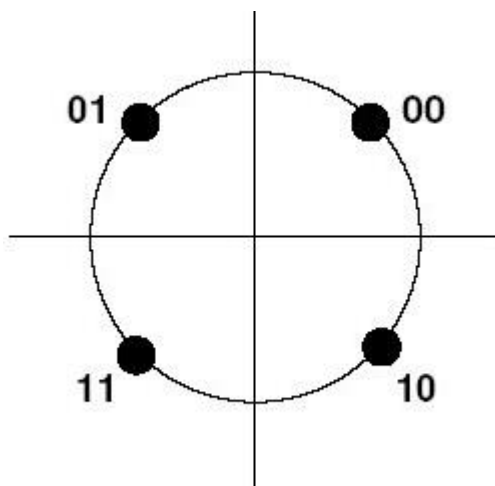
Maximální přenosová rychlost u takto modulovaného signálu je 27 Mb/s. U přenosu signálu, ale dochází ke vzniku chyb a proto se musí zakomponovat konvoluční kódování. Konvoluční kódování je samoopravný kód, což představuje prokládání přenášených dat opravnými bity, které slouží k tomu, aby nevznikla souvislá chyba.



Obr. 6.2: Konstelační diagram 64QAM

QPSK modulace nabývá 4 stavů, u nichž je amplituda stejná a fáze má 4 různé hodnoty. V tomto případě při přenosu modulované subnosné dochází k přenosu pouze 2 bitů ( $2^2 = 4$  stavy). Přenášený signál modulovaný QPSK je z trojice zminovaných metod nejméně náchylný na vznik chyb.





Obr. 6.3: Konstelační diagram QPSK

## Pracovní postup


### 1. Část úkolu: 16-QAM

- Zapněte PC, po načtení systému najděte na ploše ikonu Matlabu a dvojklikem spusťte program.

Po spuštění (spuštění může trvat delší dobu, buďte trpěliví), se dostanete do prostředí Matlab.

- V levém horním rohu v záložce **HOME** klikněte na *New -> Simulink Model*, tím spustíme prostředí Simulink (proces spuštění může také trvat chvíli déle).

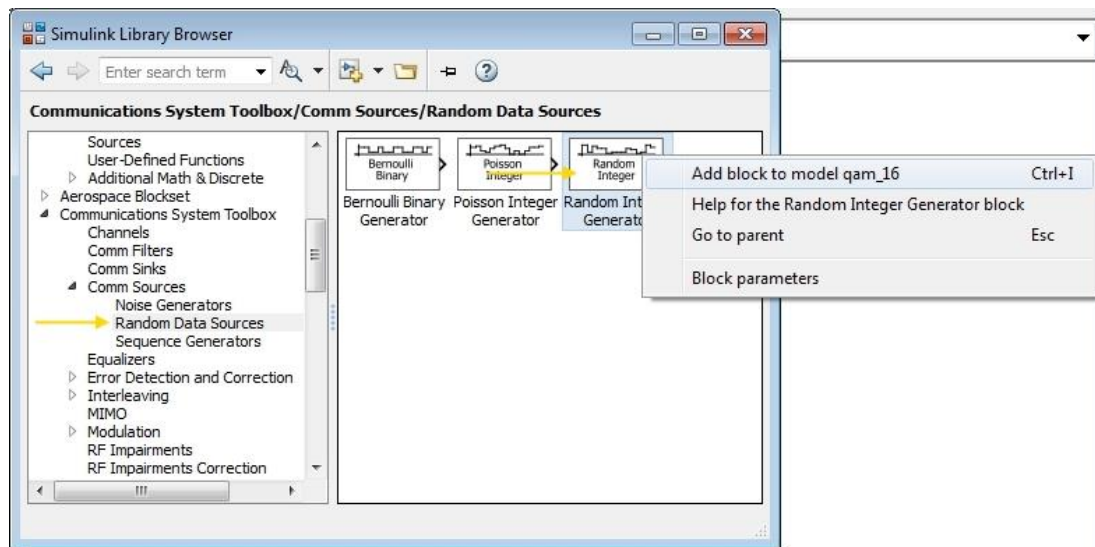
Nyní můžete přejít k sestavení a spojení celého schématu 16-QAM.

- Jako první v prostředí Simulink klikněte na *File->Save As...* a uložíte svůj první model jako **qam\_16.slx**, dále klikněte v horní nabídce na ikonu **Library Browser** 

Spustí se vám prohlížeč s vyhledávačem jednotlivých bloků z knihoven. Tímto způsobem budete přidávat všechny potřebné bloky do zapojení.

- V Library Browser v levé části rozkliknete *Communications System Toolbox -> Comm Sources -> Random Data Sources*, v pravé části se vám objeví bloky, vybereme **Random Integer Generator**, přidat jej můžete dvěma způsoby. Buď na

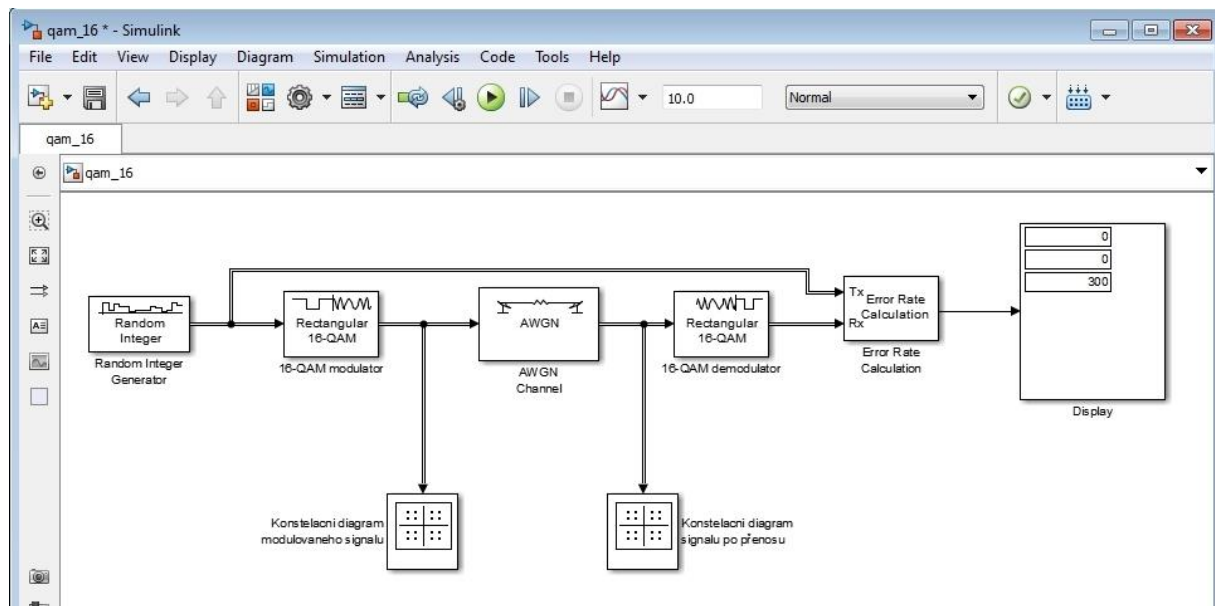
blok kliknete pravým tlačítkem myši a zvolíte *Add block to model qam\_16* (Viz. Obr. 5.4) nebo blok chytne levým tlačítkem myši a přesunete na pracovní plochu Simulinku.



Obr. 6.4: Přidání bloku

#### Dále přidejte potřebné bloky:

- *Rectangular QAM Modulator Baseband*
  - *AWGN Channel*
  - *2x Constellation Diagram*
  - *Rectangular QAM Demodulator Baseband*
  - *Error Rate Calculation*
  - *Display*
- Přidané bloky rozmístěte a spojte podle **obrázku 6.5**, bloky vhodně pojmenujte, abyste se v zapojení jednoduše orientovali.



Obr. 6.5: Zapojení 16-QAM

Nyní je třeba nastavit jednotlivé bloky, aby zapojení fungovalo a bylo možno jej simulovat. Nastavení bloků změníte dvojklikem.

#### Nastavení generátoru:

- M-ary number: *16*
- Initial seed: *1000*
- Sample time: *0.1*
- Frame-based outputs: *zaškrtnout*
- Samples per frame: *300*
- Output data type: *uint8*

- Nastavení potvrďte kliknutím na **OK**, tak potvrzujete u všech nastavení.

#### Nastavení modulátoru:

- M-ary number: *16*
- Input type: *Integer*
- Constellation ordering: *Binary*
- Normalization method: *Min. distance between symbols*

- Minimum distance: 2
- Phase offset: 0
- V záložce *Data Types* nastavte Output data type: *single*.

#### Nastavení AWGN Channel:

- Input processing: *Columns as channels (frame based)*
- Initial seed: 1000
- Mode: *Signal to noise ratio (Es/No)*
- Es/No (dB): 30
- Input signal power, referenced to 1 ohm: 1
- Symbol period:  $4 \cdot 0.1$



#### Nastavení demodulátoru:

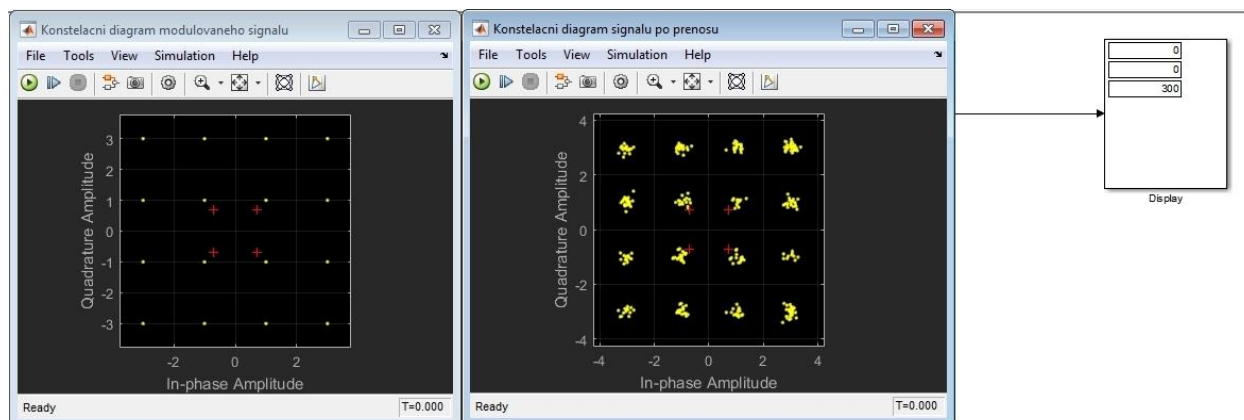
- M-ary number: 16
- Normalization method: *Min. distance between symbols*
- Minimum distance: 2
- Phase offset: 0
- Constellation ordering: *Binary*
- V záložce *Data Types* nastavte Output: *uint8*

#### Nastavení Error Rate Calculation:

- V tomto bloku změňte jen Output data na *Port*, abyste mohli připojit display k Error Rate Calculation a zobrazoval vám výsledky měření.
- Blok Display nechte v původním nastavení, jen jej trochu roztáhněte, abyste dobře viděli výsledky. První řádek na displeji zobrazuje **chybovost**, druhý řádek počet vzniknutých chyb a třetí celkový počet přenesených symbolů.

Vše byste měli mít připraveno k simulaci tohoto zapojení.

Simulaci spustíte tlačítkem *Run* , pokud jste neudělali v konfiguraci nikde chybu, zobrazí se vám výsledky simulace (Viz. Obr. 6.6). Pokud na konstelačních diagramech nevidíte všechny body, klikněte na tlačítko *Scale X & Y Axes Limits* .



Obr. 6.6: Výsledky simulace 16-QAM

Ted' zapojení otestujte tak, že budete měnit hodnotu poměru signálu k šumu (v AWGN channel snižujte hodnotu  $E_s/N_0$  (dB), až se dostanete na 100% chybovost) a sledujte změny v zobrazení bodů v konstelačním diagramu po přenosu signálu a hodnoty chybovosti z prvního řádku displeje.

- Zaznamenejte si 8 hodnot poměru signálu k šumu a k tomu 8 hodnot výsledné chybovosti. Vše uložte do Excelové tabulky, zástupce programu najdete na pracovní ploše.

## 2. Část úkolu: 64-QAM

Vaše předchozí zapojení uložte *File -> Save* a poté práci uložte jako **qam\_64.slx** *File -> Save As...*, nebudeme upravovat sestavení tohoto zapojení, ale jen nastavení jednotlivých bloků.

### Nastavení generátoru:

- Změňte nastavení M-ary number: 64

### Nastavení modulátoru:

- M-ary number: 64
- Minimum distance: 1

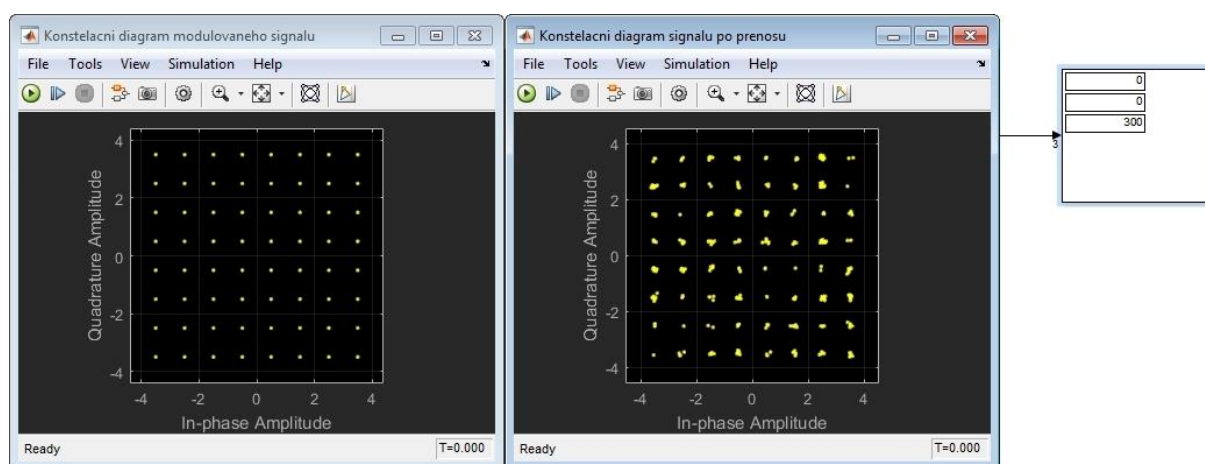
### Nastavení AWGN Channel:

- Symbol period:  $6 \cdot 0.1$

### Nastavení demodulátoru:

- M-ary number: 64
- Minimum distance: 1

Zbytek nastavení ponechte jako u předchozího zapojení. Simulaci znovu spustíte tlačítkem *Run*.

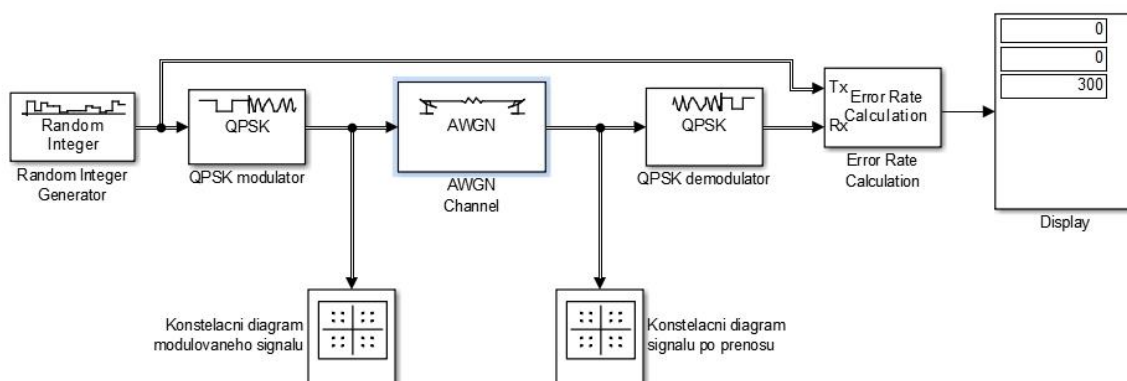


Obr. 6.7: Výsledky simulace 64-QAM

- Měňte hodnoty poměru signálu k šumu stejně, jak jste volili u 16-QAM a zapište do tabulky opět 8 hodnot chybovosti, všimněte si, že chybovost je vyšší jak u předchozí modulace.

### 3. Část úkolu: QPSK

- Vytvořte úplně nový model, uložte jej jako qpsk.slx.
- Přidejte bloky:
  - *Random Integer Generator*
  - *QPSK Modulator Baseband*
  - *AWGN Channel*
  - *2x Constellation Diagram*
  - *QPSK Demodulator Baseband*
  - *Error Rate Calculation*
  - *Display*
- Sestavte zapojení dle obrázku 6.8.



Obr. 6.8: Zapojení QPSK

- Po sestavení zapojení přejděte ke konfiguraci jednotlivých bloků.

#### Nastavení generátoru:

- M-ary number: 4
- Initial seed: 1000
- Sample time: 0.1
- Frame-based outputs: *zaškrtnout*
- Samples per frame: 300
- Output data type: *uint8*

### **Nastavení modulátoru:**

- Phase offset:  $\pi/4$
- Constellation ordering: *Binary*
- Input type: *Integer*
- V záložce Data Types nastavte Output data type: *single*

### **Nastavení AWGN:**

- Input processing: *Columns as channels (frame based)*
- Initial seed: *1000*
- Mode: *Signal to noise ratio (Es/No)*
- Es/No (dB): *30*
- Input signal power, referenced to 1 ohm: *1*
- Symbol period:  $2 \cdot 0.1$

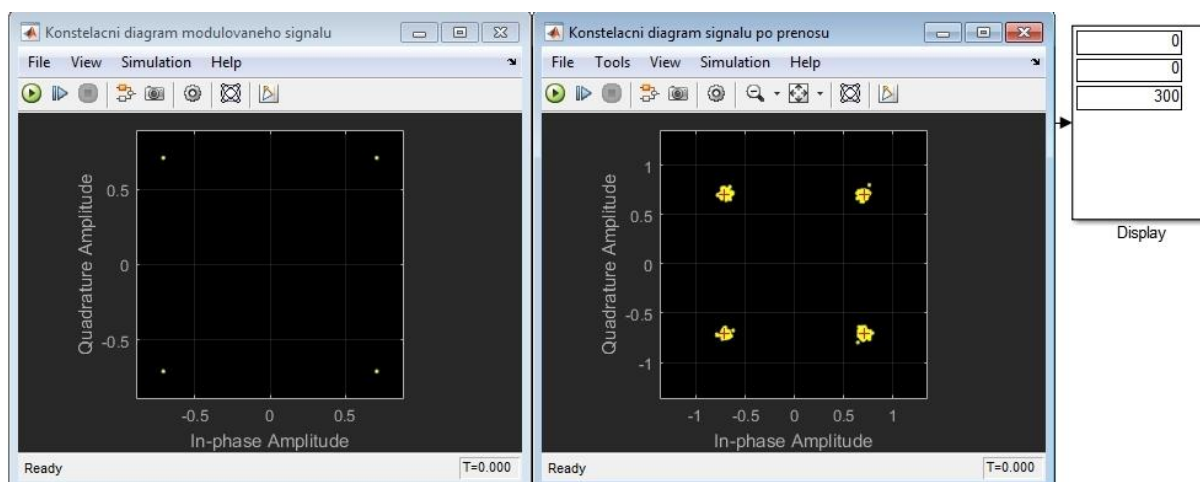
### **Nastavení demodulátoru:**

- Phase offset:  $\pi/4$
- Constellation ordering: *Binary*
- Output type: *Integer*
- V záložce Data Types nastavte Output: *uint8*

### **Nastavení Error Rate Calculation:**

- Stejně tak jako u předchozích verzí nastavte jen Output data: *Port*, abyste byli schopni připojit displej. Displej nijak neupravujte.
- Spustěte simulaci a nastavujte hodnoty signál/šum, tak jak jste zvolili u předchozích zapojení a zaznamenejte do tabulky hodnoty výsledných chybovostí z displeje.





Obr. 6.9: Výsledky simulace QPSK

Nakonec si v Excelu vytvořte graf, na kterém porovnáte závislost signál/šum na chybovost. Výsledný graf předved'te vyučujícímu, okomentujte jej a odpovězte na doplňující otázky.

## Doplňující otázky

1. U které modulace při přenosu dojde k nejméně vzniklým chybám (neberte v úvahu opravné kódování) ?
2. Která z modulací při přenosu dovoluje nejvyšší rychlost?
3. Kde se tyto modulace využívají?

## **7 Laboratorní úloha č.2**

### **Úloha č. 2 – Simulace modulace PCM**

#### **Zadání a cíl úlohy**

V této úloze se seznámíte s prostředím Matlab a Simulink. Sestavíte blokové zapojení v programu Simulink, na kterém si prakticky znázorníte jednotlivé procesy pulzní kódové modulace a ověříte si své teoretické znalosti. Zobrazené průběhy signálu zaznamenáte do protokolu laboratorní úlohy a necháte zkontrolovat vyučujícím.

1. Vytvořte blokové zapojení Pulzní kódové modulace v programu Simulink
2. Získané výsledky odevzdejte a popište vyučujícímu

#### **Vybavení pracoviště**

- PC
- program MATLAB

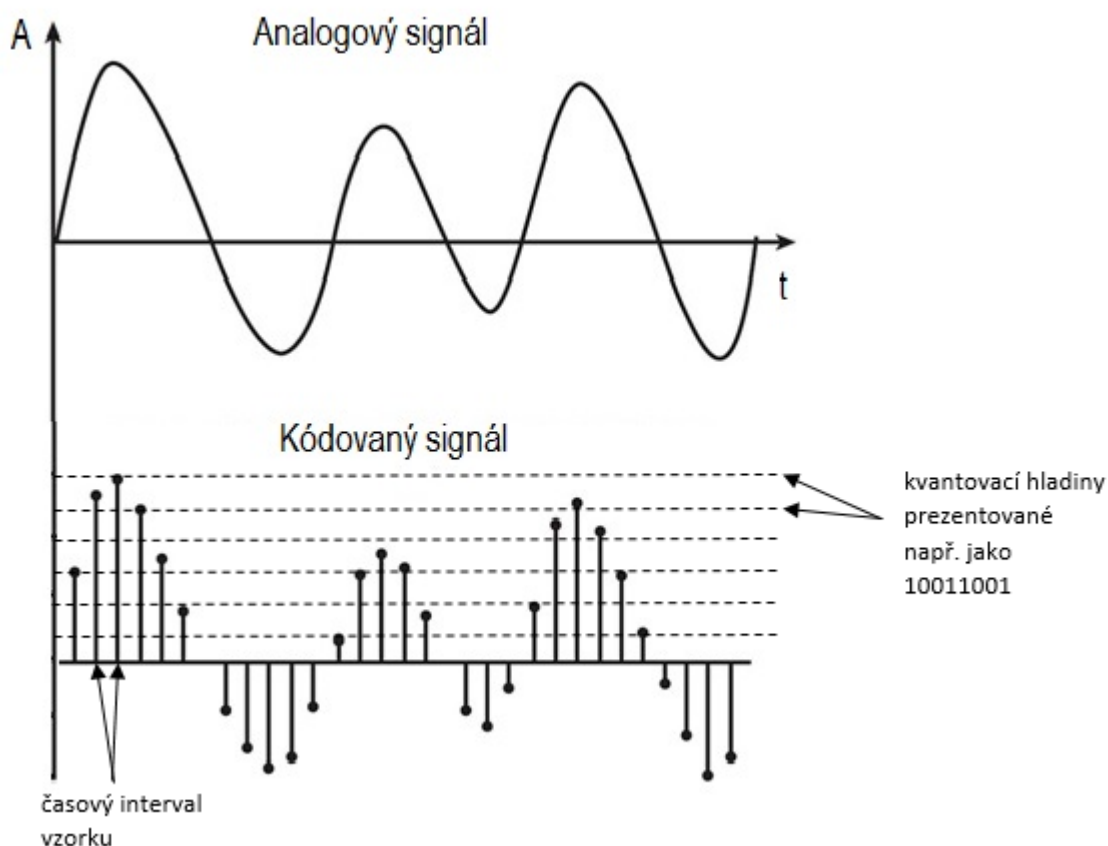
# Teoretický úvod

## Modulace PCM

Pulzní kódová modulace je digitální modulace, která se využívá pro přechod mezi analogovými a digitálními signály. Využívá se např. u hovorového přenosu nebo v hudebním průmyslu při zápisu dat na hudební CD.

Modulace spočívá ve třech základních krocích, kterými jsou: **Vzorkování**, **Kvantování** a **Kódování**. Signál se nejprve vzorkuje, čímž vznikne ze spojitého signálu diskrétní PAM signál. Takto upravený signál se nadále kvantuje, signál je rozdělen na určitý konečný počet hladin, které se nazývají kvantizační hladiny. V poslední části je třeba přidat těmto úrovním (hladinám) nějakou bitovou informaci, pomocí které se budou tyto hladiny prezentovat. K tomuto procesu dochází při kódování, kdy je jednotlivým hladinám přiřazeno kódové slovo, které se skládá ze symbolů. Jak již bylo zmíněno, toto kódové slovo je složeno ze skupiny bitů (např. 8 bitů). Tyto bity mohou nabývat hodnot buď logické 1 anebo logické 0.

Jak modulace probíhá je zjednodušeně znázorněno na Obr. 7.1



Obr. 7.1: Znázornění PCM


## Pracovní postup

- Zapněte PC, po načtení systému najdete na ploše ikonu Matlabu a dvojklikem spustíte program.

Po spuštění (spuštění může trvat delší dobu, buďte trpěliví), se dostanete do prostředí Matlab.

- V levém horním rohu v záložce **HOME** klikněte na *New -> Simulink Model*, tím spustíme prostředí Simulink (proces spuštění může také trvat chvíli déle).

Nyní můžete přejít k sestavení a spojení celého blokového zapojení PCM.

- Jako první v prostředí Simulink klikněte na *File->Save As...* a uložíte model jako **pcm.slx**, dále klikněte v horní nabídce na ikonu **Library Browser** 

Spustí se vám prohlížeč s vyhledávačem jednotlivých bloků z knihoven. Tímto způsobem budete přidávat všechny potřebné bloky do zapojení.

- V Library Browser v levé části najdeme vyhledávač, pomocí kterého budete vyhledávat jednotlivé bloky, přidat jej můžete dvěma způsoby. Buď na blok kliknete pravým tlačítkem myši a zvolíte *Add block to model pcm* nebo blok chytete levým tlačítkem myši a přesunete na pracovní plochu Simulinku.

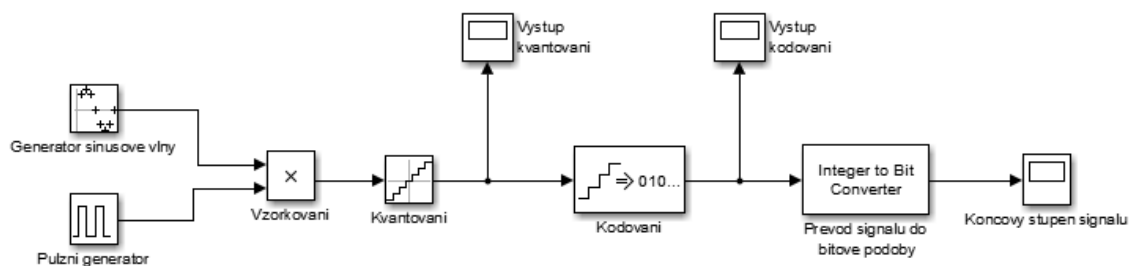
Nyní je třeba přidat všechny bloky, které k simulaci budeme potřebovat.

### Přidejte následující bloky:

- *Sine Wave*
- *Pulse Generator*
- *Product*
- *Quantizer*
- *Uniform Encoder*
- *Integer to Bit Converter*
- *3x Scope*

Po přidání bloků je třeba je uspořádat a propojit na pracovní ploše simulinku. Kliknutím na text pod blokem je lze přejmenovat. Přejmenujte je zejména z důvodu přehlednosti, vyvarujete se tak způsobením možných chyb.

- Bloky propojte podle **Obr. 7.2**.



Obr. 7.2: Blokové zapojení PCM

Nyní je třeba všechny bloky správně nakonfigurovat, abyste byli schopni model simulovat a zapojení fungovalo správně.

#### Nastavení generátoru sinusové vlny:

- Sine type: *Sample based*
- Time (t): *Use simulation time*
- Amplitude: *1*
- Bias: *0*
- Samples per period: *100*
- Number of offset samples: *0*
- Sample time: *0.1*
- Interpret vector parameters as 1-D: *Zaškrknuto*

#### Nastavení pulzního generátoru:

- Pulse type: *Time based*
- Time (t): *Use simulation time*
- Amplitude: *1*
- Period (secs): *0.1*
- Pulse Width (% of period): *1*
- Phase delay (secs): *0*
- Interpret vector parameters as 1-D: *Zaškrknuto*

### Nastavení Vzorkování:

- V záložce Main nastavte Number of inputs: 2
- V záložce Signal Attributes nastavte Integer rounding mode: *Zero*

### Nastavení kvantování:

- Quantization interval: *0.01*
- Treat as gain when linearizing: *Zaškrtnuto*


### Nastavení kódování:

- Peak: *0.1*
- Bits: 8
- Output type: *Unsigned integer*


### Nastavení převodu signálu:

- Number of bits per integer (M): 8
- Treat input values as: *Unsigned*
- Output bit order: *MSB first*
- Output data type: *Inherit via internal rule*

### Nastavení bloku Scope:

U bloku scope je nejdůležitější nastavit parametry os. Osy nastavíte pomocí tlačítka *Parameters* , který najdete po rozkliknutí scope vlevo nahoře. V záložce general upravte hodnotu *Time range*. Svislou osu nastavíte kliknutím pravého tlačítka na samotný průběh a zvolíte *Axes properties...* a nastavíte hodnotu Y-min a Y-max. Aby byl průběh dobře vidět, klikněte na tlačítko *Autoscale*, které vycentruje obraz.

Vše byste měli mít připraveno k simulaci tohoto zapojení.

- Simulaci spustíte tlačítkem *Run* . Pokud proběhne vše v pořádku, kompilace by měla projít na 100% a v dolním levém okraji by mělo být napsáno *Ready*. Pokud se tak stalo, otevřete si všechny tři výstupy signálu Výstup kvantování, Výstup kódování a Koncový stupeň signálu. Všechny tři signály si pozorně prohlédněte, zaznamenejte

do laboratorního protokolu, odpovězte na doplňující otázky a odevzdejte vyučujícímu, který vaše měření zkontroluje.

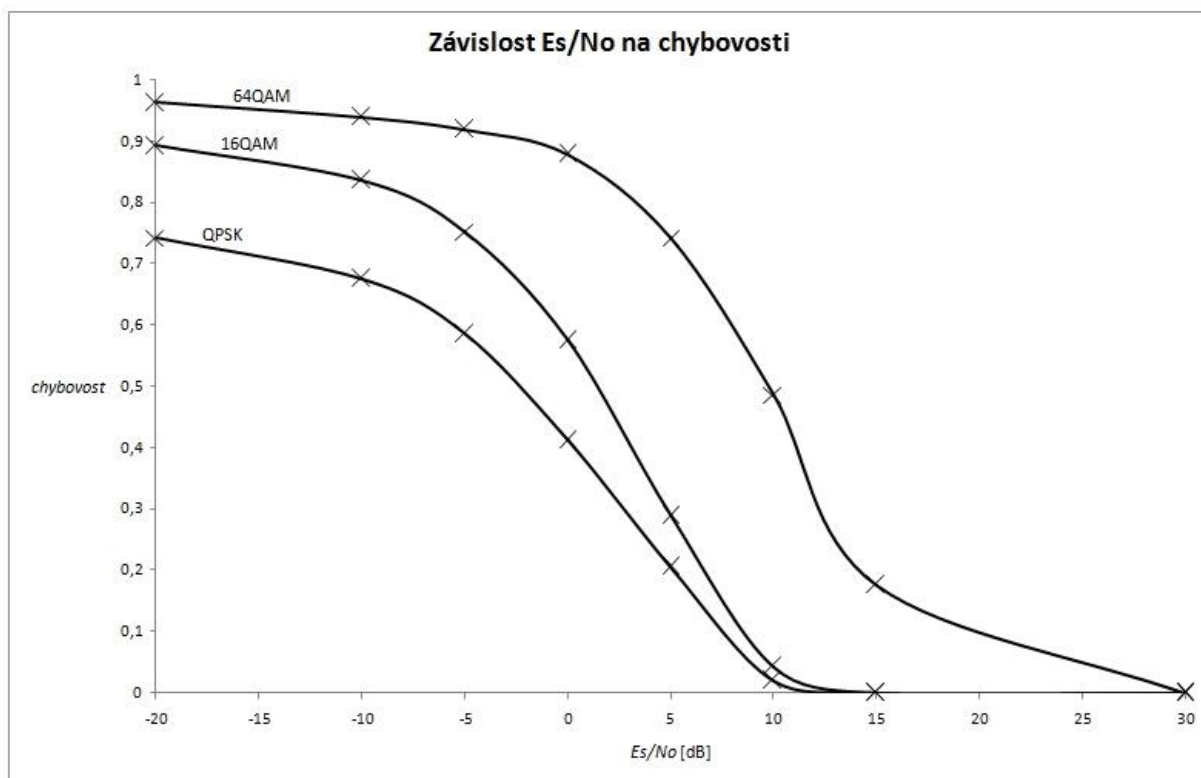
## **Doplňující otázky**

1. K čemu slouží Pulzní kódová modulace?
2. K jakým základním třem operacím dochází při modulaci?
3. Kde se nejvíce využívá PCM?

## 8 Informace pro vyučující k úlohám

### Úloha č.1

Studenti by měli bez problémů tuto úlohu podle návodu nastavit, problém by mohl nastat v přehlédnutí nějakého nastavení jednotlivých bloků. Studenti Vás požádají o kontrolu vytvořeného grafu (Viz. Obr. 8.1).



Obr. 8.1: Výsledný graf

### 8.1 Odpovědi na doplňující otázky laboratorní úlohy č.1

1. U které modulace při přenosu dojde k nejméně vzniklým chybám (neberte v úvahu opravné kódování) ?

**1. Odpověď:** QPSK

2. Která z modulací při přenosu dovoluje nejvyšší rychlost?

**2. Odpověď:** 64-QAM

3. Kde se tyto modulace využívají?

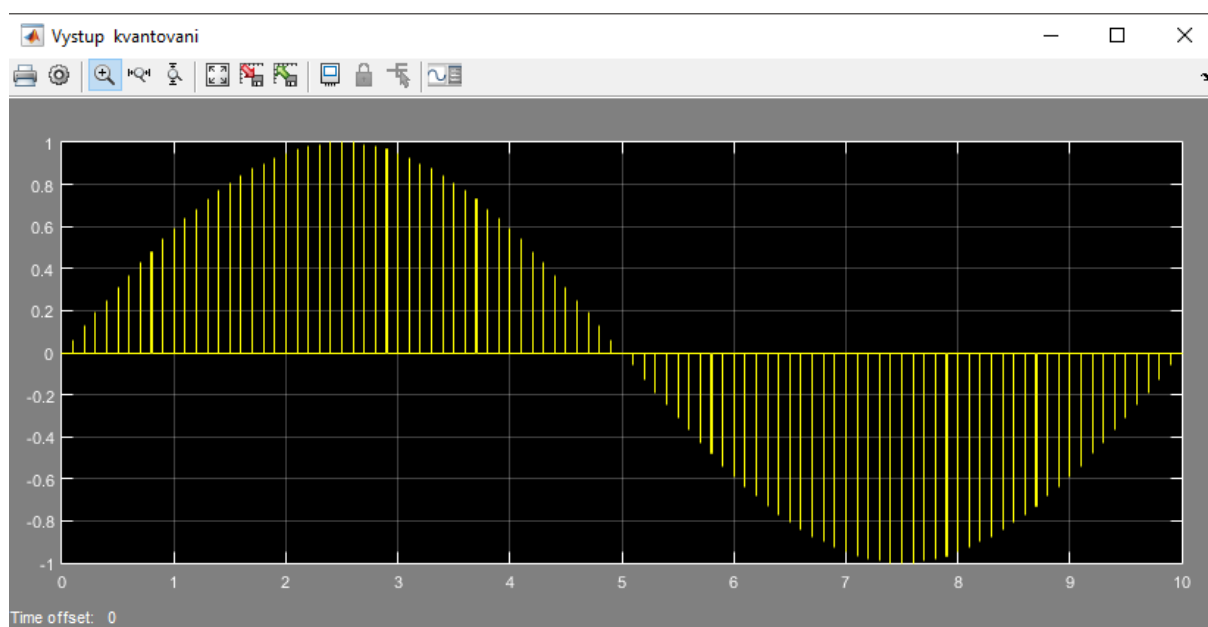


**3. Odpověď:** U kabelového vysílání je to zejména 64-QAM, u satelitního vysílání je to robustní modulace QPSK.

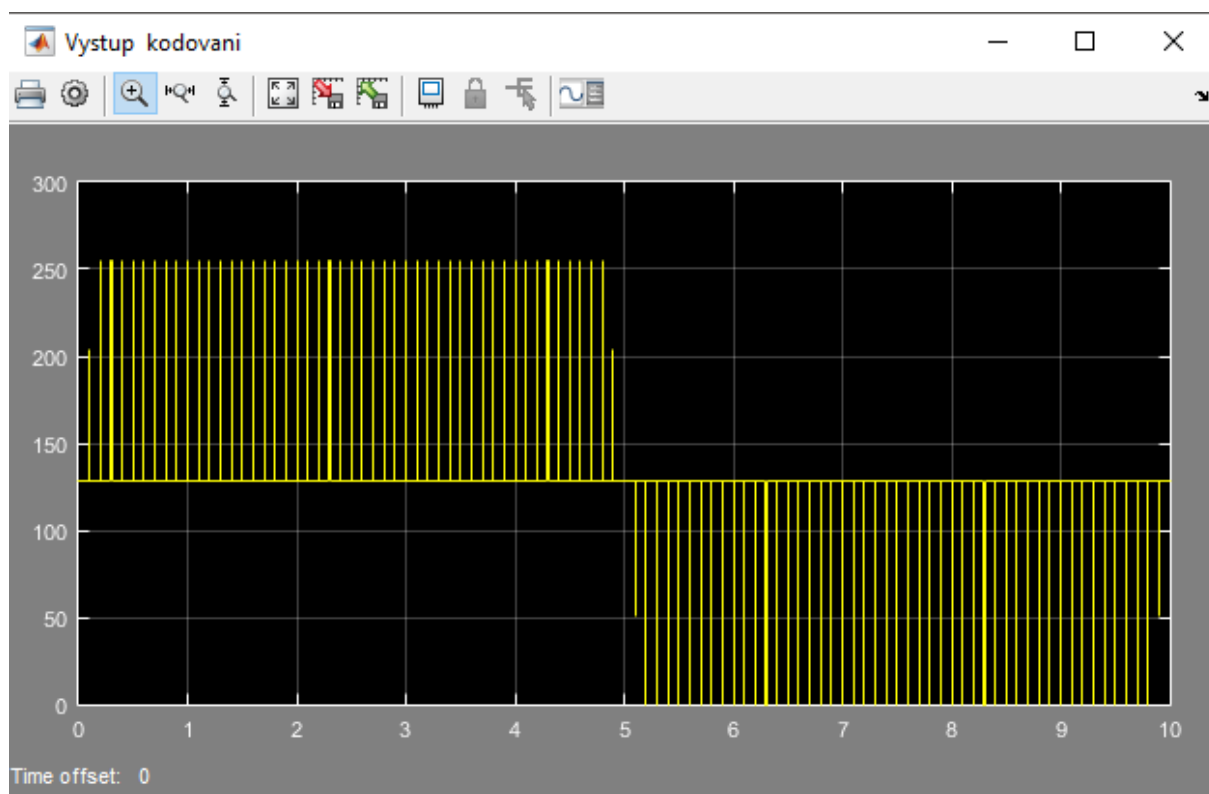
## Úloha č.2

Při vypracovávání úlohy by studenti mohli udělat chybu v případě nepozornosti. Záměnou bloku za jiný funkční blok nebo špatným nastavením blokového zapojení. Studenti vám odevzdají ke kontrole výsledky simulace a zodpovězené doplňující otázky.

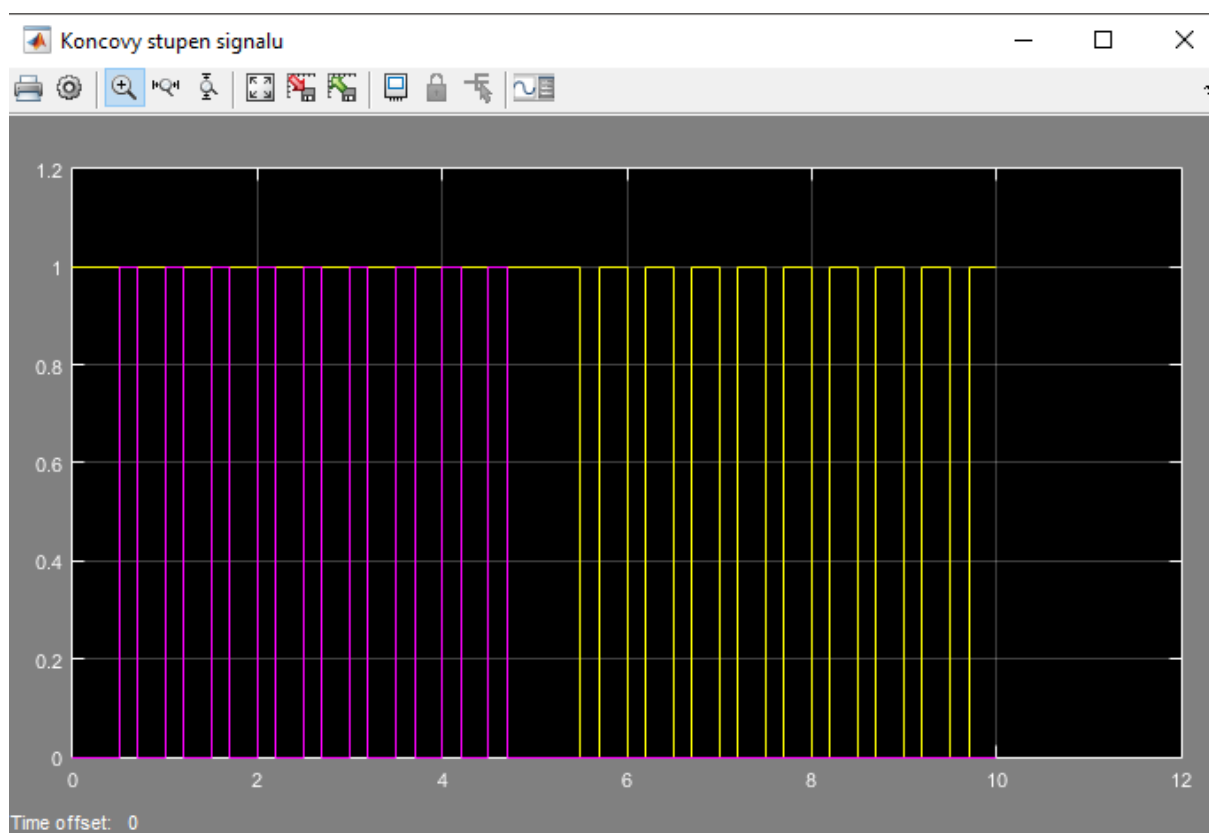
## Výsledky simulace



Obr. 8.2: Výstup kvantování



Obr. 8.3: Výstup kódování



Obr. 8.4: Koncový stupeň signálu

## 8.1 Odpovědi na doplňující otázky laboratorní úlohy č.2

1. K čemu slouží Pulzní kódová modulace?

**1. Odpověď:** K převodu analogového signálu na digitální.

2. K jakým základním třem operacím dochází při modulaci?

**2. Odpověď:** Vzorkování, Kvantování a Kódování.

3. Kde se nejvíce využívá PCM?

**3. Odpověď:** Při přenosu hlasu nebo u hudební techniky při přenosu dat na hudební CD.

## Závěr

V úvodu práce jsem krátce představil prostředí Matlab a Simulink a práci s ním, což by mělo pomoci zejména studentům, kteří se s programem ještě nesetkali. V další části je rozebrán vývoj poskytování služeb a technologií, díky kterým je možno těchto služeb využívat. K největšímu pokroku došlo zřejmě u datových komunikací, kde došlo k integraci společně s telefonii a proudového toku dat v reálném čase. To vše je nám schopna poskytnout jedna síť. V dnešní době jsou tyto služby velmi využívány, společnosti ke komunikaci mezi sebou využívají videokonference a video streaming je také využíván společnostmi k živému vysílání různých prezentací.

Pro vytvoření prvního modelu v prostředí Simulink jsem zvolil z vysokorychlostních digitálních televizních sítí modulaci QAM a QPSK. U těchto kvadraturních vícecestavových modulací lze simulovat modulaci, přenos a demodulaci signálu. Do laboratorní úlohy jsem vybral modulace 16-QAM, 64-QAM a QPSK, které se hojně využívají u kabelového přenosu televizního vysílání a satelitního přenosu. Studenti si názorně zobrazí jednotlivé konstelační diagramy všech těchto typů modulací a ověří si teoretické znalosti ohledně chybovosti závislé na šumu. Tyto hodnoty zpracují a vytvoří graf, který předvedou svému vyučujícímu a okomentují jej.

V druhé laboratorní úloze jsem použil model pulzní kódové modulace PCM. Jedná se o digitální modulaci, která slouží k převodu spojitého analogového signálu na signál digitální. Tato modulace je nejvíce využívána u hovorových přenosů, kde je na vstupu analogový signál a je třeba jej modulací upravit pro přenos. Taktéž se využívá při přenosu dat na hudební CD, používají různé kvality záznamu zvuku. V laboratorní úloze si studenti ověří své teoretické znalosti pomocí již zmiňovaného modelu PCM. Po simulaci zapojení si zobrazí výstupy z jednotlivých kroků Vzorkování, Kvantování a Kódování. Výsledky zaznamenají a odevzdají vyučujícímu.

V závěru práce se nacházejí informace pro vyučující, které obsahují všechny výsledky laboratorních úloh, pomocí kterých mohou zkontrolovat jejich korektnost zpracování.

# Literatura

1. ZAPLATÍLEK, Karel a Bohuslav DOŇAR. *MATLAB: tvorba uživatelských aplikací*. 1. vyd. Praha: BEN - technická literatura, 2004, 215 s. ISBN 80-7300-133-0.
2. ZAPLATÍLEK, Karel a Bohuslav DOŇAR. *MATLAB: začínáme se signály*. 1. vyd. Praha: BEN - technická literatura, 2006, 271 s. ISBN 80-7300-200-0.
3. PUŽMANOVÁ, Rita. *Moderní komunikační sítě od A do Z: [technologie pro datovou, hlasovou i multimediální komunikaci]*. 2., aktualiz. vyd. Brno: Computer Press, 2006, 430 s. ISBN 80-251-1278-0.
4. POOLE, Ian. . *Radio-Electronics*. [online]. [cit. 2015-10-11]. Dostupné z: <http://www.radio-electronics.com/info/rf-technology-design/quadrature-amplitude-modulation-qam/8qam-16qam-32qam-64qam-128qam-256qam.php>
5. ŘÍČNÝ, Václav. *Televizní distribuční sítě a systémy: přednášky a numerická cvičení*. Vyd. 1. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav radioelektroniky, 2005, 82 s. ISBN 80-214-2816-3.
6. HANUŠ, Petr. *Matlab – Úvod*. . [online]. [cit. 2015-10-10]. Dostupné z: <http://mat.fsv.cvut.cz/hanus/matlab/default.html>
7. The MathWorks, Inc.. *Matlab. Mathworks*. [online]. © 1994-2015 [cit. 2015-10-10]. Dostupné z: <http://www.mathworks.com/products/matlab/>
8. KOLÁČEK, Jan a Kateřina KONEČNÁ. *Jak pracovat s MATLABem* [online]. . [cit. 2015-11-12]. ISBN . Dostupné z: <https://www.math.muni.cz/~kolacek/vyuka/vypsyst/navod.pdf>
9. ŠKORPIL, Vladislav. *Vysokorychlostní komunikační systémy*. Brno, 2013. ISBN 978-80-214-4718-9.
10. HUMUSOFT. *Simulink – Simulace a modelování dynamických systémů*. . [online]. 1991 – 2015 [cit. 2015-11-12]. Dostupné z: <http://www.humusoft.cz/matlab/simulink/>
11. SZABÓ, Zoltán . *Programování v Matlabu*. . [online]. 2007 [cit. 2015-11-25]. Dostupné z: <http://webzam.fbmi.cvut.cz/szabozol/matlab/cv3.pdf>
12. HÉDL, Lukáš. *Popis systému Matlab*. . [online]. 2003 [cit. 2015-11-25]. Dostupné z: [http://www.352.vsb.cz/uc\\_texty/AutoRizMatlab/kap1.html](http://www.352.vsb.cz/uc_texty/AutoRizMatlab/kap1.html)
13. *Simulink®: simulation and model-based design : using Simulink*. Version 6. Natick, MA: MathWorks, 2005, 1 v. (various pagings).
14. PETERKA, Jiří. *Co je konvergence?*. . [online]. 2000 [cit. 2015-12-01]. Dostupné z: <http://www.earchiv.cz/b00/b1200002.php3>

15. PUŽMANOVÁ, Rita. Vývoj paketových sítí a postavení MPLS. *Svět sítí*. [online]. 2006 [cit. 2015-12-01]. Dostupné z: <http://www.svetsiti.cz/clanek.asp?cid=Vyvoj-paketovych-siti-a-postaveni-MPLS-2472006>
16. ŘÍČNÝ, Václav. *Televizní distribuční sítě a systémy: přednášky a numerická cvičení*. Vyd. 1. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav radioelektroniky, 2005, 82 s. ISBN 80-214-2816-3.
17. NOSTERSKÝ, František. *Elektronická Zařízení* [online]. 2009. [cit. 2015-12-05].
18. ŠTEFEK, Petr. svetaudia. . [online]. 21.8.2014 [cit. 2016-03-15]. Dostupné z: <http://www.svetaudia.cz/dsd-spasa-nebo-marketing-hudebniho-prumyslu/66-2>
19. ŽALUD, Václav. *Moderní radioelektronika*. Praha: BEN - technická literatura, 2000. ISBN 80-86056-47-3.
20. MIŠENČÍK, Patrik. Analýza vybraných pulzních modulačních metod . . [online]. 2011 [cit. 2016-03-20]. Dostupné z: [https://dk.upce.cz/bitstream/handle/10195/41813/MisencikP\\_AnalyzaVybranych\\_JP\\_2011.pdf?sequence=2&isAllowed=y](https://dk.upce.cz/bitstream/handle/10195/41813/MisencikP_AnalyzaVybranych_JP_2011.pdf?sequence=2&isAllowed=y)

## Seznam obrázků

Obr. 1.1: Ukázka jednoduchého výpočtu a zobrazení sinusového průběhu.....	9
Obr. 1.2: Ukázka práce se souborem m-file.....	10
Obr. 1.3: Ukázka prostředí a knihovny Simulink.....	11
Obr. 1.4: Druhá možnost zobrazení prostředí Simulink.....	12
Obr. 3.1: ATM buňka.....	16
Obr. 5.1: Blokové schéma PCM.....	19
Obr. 6.1: Konstelační diagram 16QAM.....	23
Obr. 6.2: Konstelační diagram 64QAM.....	24
Obr. 6.3: Konstelační diagram QPSK.....	25
Obr. 6.4: Přidání bloku.....	26
Obr. 6.5: Zapojení 16-QAM.....	27
Obr. 6.6: Výsledky simulace 16-QAM.....	29
Obr. 6.7: Výsledky simulace 64-QAM.....	30
Obr. 6.8: Zapojení QPSK.....	31
Obr. 6.9: Výsledky simulace QPSK.....	33
Obr. 7.1: Znázornění PCM.....	35
Obr. 7.2: Blokové zapojení PCM.....	37
Obr. 8.1: Výsledný graf.....	40
Obr. 8.2: Výstup kvantování.....	41
Obr. 8.3: Výstup kódování.....	42
Obr. 8.4: Koncový stupeň signálu.....	42

## Seznam tabulek

Tab. 4.1: Rozdělení modulací.....	18
Tab: 5.1 Shrnutí využívaných modulací u hudebních CD.....	21